

1 What is claimed is:

Sub  
C1  
1. A method for preventing access to a shared peripheral device by a processor-based node in a multinode system, including the steps of:

5 (1) storing at the peripheral device a first unique value representing a first configuration of the multinode system;

(2) sending an access request from the node to the device, the request including a second unique value representing a second configuration of the multi-node system;

(3) determining whether said first and second values are identical; and

10 (4) if the first and second values are identical, then executing the access request at the peripheral device.

2. The method of claim 1, wherein:

15 said first value is generated utilizing at least in part information relating to a first time when the multinode system was in said first configuration; and

said second value is generated utilizing at least in part information relating to a second time when the multinode system was in said second configuration.

3. The method of claim 2, wherein:

20 step 3 includes the step of determining whether said first and second times are identical.

4. The method of claim 1, wherein said first and second values are generated based at least in part on epoch numbers generated by a membership protocol executing on said multinode system.

25 5. The method of claim 4, wherein each of said first and second values is generated based at least in part on respective membership sets of said multinode system generated by said membership protocol.

30 6. The method of claim 1, wherein each of said first and second values is generated based at least in part on respective membership sets of said multinode system generated by said membership protocol.

1  
7. An apparatus for preventing access to at least one shared peripheral resource by a processor-based node in a multinode system, the resource being coupled to the system by a resource controller including a controller memory, each of a plurality of nodes on the system including a processor coupled to a node memory storing program modules configured to executing functions of the invention, the apparatus including:

5 a membership monitor module configured to determine a membership list of the nodes, including said resource, on the system at predetermined times, including at least at a time when the membership of the system changes;

10 a resource manager module configured to determine when the resource is in a failed state and for communicating the failure of the resource to said membership monitor to indicate to the membership monitor to generate a new membership list;

15 a configuration value module configured to generate a unique value based upon said new membership list and to store said unique value locally at each node on the system; and

an access control module stored at said controller memory configured to block access requests by at least one said requesting node to said resource when the locally stored unique value at said requesting node does not equal the unique value stored at said resource controller.

20 8. The apparatus of claim 7, wherein said configuration value monitor module is configured to determine said unique value based at least in part upon a time stamp indicating the time at which the corresponding membership list was generated.

25 9. The apparatus of claim 7, wherein said unique value is based at least in part upon an epoch number generated by a membership protocol module.

10. The apparatus of claim 7, wherein said membership monitor module is configured to execute independently of any action by said shared resource when said shared resource is in a failed state.

30 11. The apparatus of claim 7, wherein said resource manager module is configured to execute independently of any action by said shared resource when said shared resource is in a failed state.

1

12. The apparatus of claim 7, wherein said configuration module is configured to execute independently of any action by said shared resource when said shared resource is in a failed state.

5

13. The apparatus of claim 7, wherein said access control module is configured to execute independently of any action by said shared resource when said shared resource is in a failed state.

add  
ca

10

15

20

25

30